An Extended Cubic B-spline Finite Element Method for Solving Generalized Burgers-fisher Equation

Ozlem Ersoy Hepson

Department of Mathematics & Computer, Eskisehir Osmangazi University, 26480, Eskisehir, Turkey.

December 13, 2016

Abstract

Keywords: collocation methods, extended cubic B-spline, Burgers-Fisher equation.

PACS:

1 Introduction

Many numerical method uses the basis functions to solve the differential eauations. One of the widely -used basis function are B-splines which are used for setting up function approximation, computed aided-design and solutions of the differential equations. There exist B-spline based numerical methods in the numerical analysis field. New variants of spline functions have been developed to form better approximation for numerical methods. The extended B-spline functions(EBF), introduced by Han and Liu[1, 3], consist of adding additional terms to the existing B-spline functions. Additional terms include free parameters which causes to obtain different shapes of the B-spline form. EBF have the same continuity with its standart B-spline functions and is a piecewise polynomial function of degree 4. In our study we emply the collocation method to solve the generalized Burgers-Fisher equation(GBFE). The approximation function in the collocation method will be consist of the combination of theEBF over the problem domain. We will observe the accuracy of the numerical solutions when the free parameter is changed. The effect of present method will be sought for solutions of GBFE. Recently, EBF has started to form the numerical methods to solve differential equations. Numerical solutions of ordinary differential equations in the form of linear two-point boundary value problems[4, 6, 7] are given by the EBF collocation method. The method of the collocation based on the EBF is described for solving a one-dimensional heat equation with a nonlocal initial condition, Newell Whitehead Segel type equation, Modified Reqularized Long Wave equation and Advection-Diffusion equation in the works [5, 9, 10, 11]. An extended modified cubic

B-Spline differential quadrature method is proposed to approximate the solution of the nonlinear Burgers' equation[13].
This study aims to construct an algorithm of combination of Crank-Nicolson and finite element method based on extended B-spline functions to the solutions of some initial boundary value problems defined for the generalized Burgers-Fisher equation of the form

$$u_t + \alpha u^q u_x - \mu u_{xx} = \eta u(1-u^q),\ x \in [0,1],\ t \geq 0 \tag{1}$$

with the initial condition

$$u(x,0) = u_0(x) \tag{2}$$

and the boundary conditions

$$u(0,t) = \zeta_1(t),\ \ t \geq 0 \tag{3}$$

and

$$u(1,t) = \zeta_2(t),\ \ t \geq 0. \tag{4}$$

where α, η and q are parameters. The equation is a general form form both Burgers' and Fisher equations. The solution profiles of the equation covers many different types like single and multiple soliton solutions.

Studies are going on finding both analytical and numerical solutions of BFE. Due to including nonlinearity in the BFE, effective numerical algorithm are necessary to understand some physical phenomena related to the BFE. Various numerical approaches have also been implemented to the solutions of BFE. In this paper we only mention spline related numerical scheme.The cubic B-spline quasi-interpolation combined with low order finite difference were suggested to solve Burgers-Fisher equation numerically. A numerical method based on exponential spline and finite difference approximations is developed to solve the generalized Burgers'-Fisher equation[14]. The classical polynomial B-splines of cubic degree were used as basis to develop a collocation method for the numerical solutions of the Burgers-Fisher equation[15]. In [16] generalized Burgers Fisher equation have been solved numerically by way of the exponential cubic B-spline collocation method.

2 Extended Cubic B-spline Collocation Method

The blending function of the extended cubic uniform B-spline with degree 4, $E_i(x)$, can be defined as [3]

$$E_i(x) = \frac{1}{24h^4}\begin{cases} 4h(1-\lambda)(x-x_{i-2})^3 + 3\lambda(x-x_{i-2})^4, & [x_{i-2},x_{i-1}]\,, \\ (4-\lambda)h^4 + 12h^3(x-x_{i-1}) + 6h^2(2+\lambda)(x-x_{i-1})^2 & \\ -12h(x-x_{i-1})^3 - 3\lambda(x-x_{i-1})^4 & [x_{i-1},x_i]\,, \\ (4-\lambda)h^4 - 12h^3(x-x_{i+1}) + 6h^2(2+\lambda)(x-x_{i+1})^2 & \\ +12h(x-x_{i+1})^3 - 3\lambda(x-x_{i+1})^4 & [x_i,x_{i+1}]\,, \\ 4h(\lambda-1)(x-x_{i+2})^3 + 3\lambda(x-x_{i+2})^4, & [x_{i+1},x_{i+2}]\,, \\ 0 & \text{otherwise.} \end{cases} \tag{5}$$

In relation (5), the free parameter λ is used to obtain different form of extended cubic B-Spline functions. Note that when $\lambda = 0$, the basis function reduces to that of the cubic uniform B-spline. Graph of the extended cubic B-splines over the interval $[0, 1]$ is depicted in Fig. 1-2 for $\lambda = -1$, -0.5, 0, 0.5, 1 and $\lambda = -10$, -5, 0, 5, 10 respectively.

Fig.1: Extended cubic B-splines over the interval $[0, 1]$ for $\lambda = -1, -0.5, 0, 0.5, 1$

Fig.2: Extended cubic B-splines over the interval $[0,1]$ for $\lambda = \lambda = -10, -5, 0, 5, 10$

$\{E_{-1}(x), E_0(x), \cdots, E_{N+1}(x)\}$ forms a basis for the functions defined over the interval $[a,b]$. Each basis function $E_i(x)$ is twice continuously differentiable. The values of $E_i(x), E_i^{'}(x),\ E_i^{''}(x)$ at the nodal points x_i 's computed from Eq.(5) are shown Table 1.

Table 1: Values of $E_i(x)$ and its principle two derivatives at the knot points

x	x_{i-2}	x_{i-1}	x_i	x_{i+1}	x_{i+2}
$24E_i$	0	$4-\lambda$	$16+2\lambda$	$4-\lambda$	0
$2hE_i^{'}$	0	-1	0	1	0
$2h^2E_i^{''}$	0	$2+\lambda$	$-4-2\lambda$	$2+\lambda$	0

Suppose that the problem domain is $[a,b]$ is divided by the knots

$$\pi : a = x_0 < x_1 < \ldots < x_N = b$$

into elements $[x_i, x_{i+1}]$, $i = 0, 1, ..., N-1$ and with mesh spacing $h = x_{i+1} - x_i = (b-a)/N$, $i = 0, 1, ..., N-1$. Then the approximate solution U to the unknown u is written in terms of the expansion of the extended cubic B-Spline as

$$U(x,t) = \sum_{i=-1}^{N+1} \delta_i E_i(x) \tag{6}$$

where δ_i are the unknown real constants and $E_i(x)$ are the basis function of the extended cubic uniform B-spline. The nodal values U and its first and second derivatives at the knots can be found from the (6) as

$$\begin{aligned} U_i &= U(x_i,t) = \frac{4-\lambda}{24}\delta_{i-1} + \frac{8+\lambda}{12}\delta_i + \frac{4-\lambda}{24}\delta_{i+1}, \\ U_i' &= U'(x_i,t) = \frac{-1}{2h}\left(\delta_{i-1} - \delta_{i+1}\right) \\ U_i'' &= U''(x_i,t) = \frac{2+\lambda}{2h^2}\left(\delta_{i-1} - 2\delta_i + \delta_{i+1}\right) \end{aligned} \tag{7}$$

The Crank–Nicolson scheme is used to discretize time variables of the unknown U in the GBFE equation so that one obtain the time discretized form of the equation as

$$\frac{U^{n+1} - U^n}{\Delta t} = -\alpha\frac{(U^qU_x)^{n+1} + (U^qU_x)^n}{2} + \mu\frac{U_{xx}^{n+1} + U_{xx}^n}{2} + \eta\frac{U^{n+1} + U^n}{2} - \eta\frac{\left(U^{q+1}\right)^{n+1} + \left(U^{q+1}\right)^n}{2} \tag{8}$$

where $U^{n+1} = U(x, t^{n+1})$ is the solution of the equation at the $(n+1)$th. time level. Here $t^{n+1} = t^n + \Delta t$ and Δt is the time step, superscripts denote n th time level , $t^n = n\Delta t$.

The nonlinear term $(U^qU_x)^{n+1}$ and $\left(U^{q+1}\right)^{n+1}$ in Eq. (8) is linearized by using the following form [2]:

$$(U^qUx)^{n+1} = (U^q)^n U_x^{n+1} + q\left(U^{q-1}\right)^n U_x^n U^{n+1} - q\left(U^q\right)^n U_x^n$$

and

$$\begin{aligned} (U^{q+1})^{n+1} &= (U^qU)^{n+1} \\ &= (U^q)^n U^{n+1} + q\left(U^{q-1}\right)^n U^n U^{n+1} - q\left(U^q\right)^n U^n \\ &= (1+q)\left(U^q\right)^n U^{n+1} - q\left(U^{q+1}\right)^n \end{aligned}$$

So Eq. (8) is discretized in time as

$$\begin{aligned} & U^{n+1} + \alpha\frac{\Delta t}{2}(L_1)^q U_x^{n+1} + \alpha\frac{\Delta t}{2}q(L_1)^{q-1}L_2U^{n+1} - \mu\frac{\Delta t}{2}U_{xx}^{n+1} - \eta\frac{\Delta t}{2}U^{n+1} \\ & +\eta\frac{\Delta t}{2}(1+q)(L_1)^q U^{n+1} \\ = \; & U^n - \alpha\frac{\Delta t}{2}(1-q)(L_1)^q U_x^n + \mu\frac{\Delta t}{2}U_{xx}^n + \eta\frac{\Delta t}{2}U^n - \eta\frac{\Delta t}{2}(1-q)(L_1)^q U^n \end{aligned} \tag{9}$$

Substitution (7) into (9) leads to the fully-discretized equation:

$$\begin{aligned}
&\left[\left(1+\alpha\frac{\Delta t}{2}q(L_1)^{q-1}L_2-\eta\frac{\Delta t}{2}+\eta\frac{\Delta t}{2}(1+q)(L_1)^q\right)\alpha_1+\alpha\frac{\Delta t}{2}(L_1)^q\beta_1-\mu\frac{\Delta t}{2}\gamma_1\right]\delta_{i-1}^{n+1}\\
&+\left[\left(1+\alpha\frac{\Delta t}{2}q(L_1)^{q-1}L_2-\eta\frac{\Delta t}{2}+\eta\frac{\Delta t}{2}(1+q)(L_1)^q\right)\alpha_2-\mu\frac{\Delta t}{2}\gamma_2\right]\delta_{i}^{n+1}\\
&+\left[\left(1+\alpha\frac{\Delta t}{2}q(L_1)^{q-1}L_2-\eta\frac{\Delta t}{2}+\eta\frac{\Delta t}{2}(1+q)(L_1)^q\right)\alpha_1-\alpha\frac{\Delta t}{2}(L_1)^q\beta_1-\mu\frac{\Delta t}{2}\gamma_1\right]\delta_{i+1}^{n+1}\\
&=\left[\left(1+\eta\frac{\Delta t}{2}-\eta\frac{\Delta t}{2}(1-q)(L_1)^q\right)\alpha_1-\alpha\frac{\Delta t}{2}(1-q)(L_1)^q\beta_1+\mu\frac{\Delta t}{2}\gamma_1\right]\delta_{i-1}^{n}\\
&+\left[\left(1+\eta\frac{\Delta t}{2}-\eta\frac{\Delta t}{2}(1-q)(L_1)^q\right)\alpha_2+\mu\frac{\Delta t}{2}\gamma_2\right]\delta_{i}^{n}\\
&+\left[\left(1+\eta\frac{\Delta t}{2}-\eta\frac{\Delta t}{2}(1-q)(L_1)^q\right)\alpha_1+\alpha\frac{\Delta t}{2}(1-q)(L_1)^q\beta_1+\mu\frac{\Delta t}{2}\gamma_1\right]\delta_{i+1}^{n}
\end{aligned} \tag{10}$$

where

$$\begin{aligned}
L_1 &= \alpha_1\delta_{i-1}^n+\alpha_2\delta_i^n+\alpha_1\delta_{i+1}^n\\
L_2 &= \beta_1\delta_{i-1}^n-\beta_1\delta_{i+1}^n
\end{aligned}$$

$$\begin{aligned}
\alpha_1 &= \frac{4-\lambda}{24},\ \alpha_2=\frac{8+\lambda}{12}\\
\beta_1 &= -\frac{1}{2h},\ \gamma_1=\frac{2+\lambda}{2h^2},\ \gamma_2=-\frac{4+2\lambda}{2h^2}
\end{aligned}$$

A linear system of $(N+1)$ equations in $N+3$ unknown is obtained. However, two additional linear equations are needed to obtain the values of $n+3$ unknown parameters $\mathbf{d}^{n+1}=(\delta_{-1}^{n+1},\delta_0^{n+1},\ldots,\delta_{N+1}^{n+1})$. To make solvable the system, boundary conditions $U_0=\zeta_1$, $U_N=\zeta_2$ are used to find two additional linear equations:

$$\begin{aligned}
\delta_{-1} &= \frac{1}{\alpha_1}\left(U_0-\alpha_2\delta_0-\alpha_3\delta_1\right),\\
\delta_{N+1} &= \frac{1}{\alpha_3}\left(U_n-\alpha_1\delta_{N-1}-\alpha_2\delta_N\right).
\end{aligned} \tag{11}$$

(11) can be used to eliminate δ_{-1}, δ_{N+1} from the system (10) which then becomes the solvable matrix equation. A variant of Thomas algorithm is used to solve the system.

Before starting the iteration of the Eq. (10), initial parameters $\delta_{-1}^0,\delta_0^0,\ldots,\delta_{N+1}^0$ must be determined from the initial condition and first space derivative of the initial conditions at the boundaries as the following:

1. $U(x_i,0)=U(x_i,0)$, $i=0,...,N$
2. $(U_x)(x_0,0)=U(x_0)$
3. $(U_x)(x_N,0)=U(x_N)$.

3 Numerical tests

In this section, some numerical solutions of the GBFE with the extended cubic B-Spline collocation are presented. To show the efficiency of the present method for our problem in comparison with the exact solution we report maximum error which is defined by

$$L_\infty = |u - U|_\infty = \max_j \left| u_j - U_j^n \right|$$

where U is the solution obtained by Eq. (1), solved by the extended cubic B-spline collocation method and u is the exact solution.

3.1 Example 1

Consider GBFE with the initial condition

$$u(x,0) = \left\{ \frac{1}{2} + \frac{1}{2}\tanh\left(\frac{-\alpha q}{2(q+1)}\right)x \right\}^{\frac{1}{q}} = \varphi(x)$$

the boundary conditions

$$u(0,t) = \left(\frac{1}{2} + \frac{1}{2}\tanh\left[\frac{-\alpha q}{2(q+1)} \left(-\left(\frac{\alpha}{q+1} + \frac{\eta(q+1)}{\alpha} \right) t \right) \right] \right)^{\frac{1}{q}} = \zeta_1(t),\ t \geq 0$$

and

$$u(1,t) = \left(\frac{1}{2} + \frac{1}{2}\tanh\left[\frac{-\alpha q}{2(q+1)} \left(1 - \left(\frac{\alpha}{q+1} + \frac{\eta(q+1)}{\alpha} \right) t \right) \right] \right)^{\frac{1}{q}} = \zeta_2(t),\ t \geq 0$$

its exact solution

$$u(x,t) = \left(\frac{1}{2} + \frac{1}{2}\tanh\left[\frac{-\alpha q}{2(q+1)} \left(x - \left(\frac{\alpha}{q+1} + \frac{\eta(q+1)}{\alpha} \right) t \right) \right] \right)^{\frac{1}{q}},\ t \geq 0$$

We run the program for three sets of parameters to make comparison with results of some earlier studies [12, 17, 18, 19] .

We show discrete L_∞ error norms for, $\alpha = 0.1$, $\eta = -0.0025$, $\Delta t = 0.0001$, $N = 16$ and results are documented in Table 2. Absolute error of solution at $t = 1$ is depicted in Fig. 3 when $q = 1$,

Table 2: L_∞ error norms				
	Time(t)	$l=0$	($l=-0.000003$)	[12]
$q=1$	0.1	1.08646×10^{-12}	1.02251×10^{-13}	1.32396×10^{-11}
	0.2	1.46944×10^{-12}	1.24456×10^{-13}	1.78026×10^{-11}
	0.3	1.61926×10^{-12}	1.24456×10^{-13}	1.94258×10^{-11}
	0.4	1.67277×10^{-12}	1.24456×10^{-13}	2.00083×10^{-11}
	0.5	1.67277×10^{-12}	1.24456×10^{-13}	2.02158×10^{-11}
$q=2$	0.1	2.17542×10^{-11}	1.34003×10^{-13}	2.84700×10^{-10}
	0.2	3.02457×10^{-11}	1.47881×10^{-13}	3.87950×10^{-10}
	0.3	3.33861×10^{-11}	1.47881×10^{-13}	4.24646×10^{-10}
	0.4	3.45414×10^{-11}	1.47104×10^{-13}	4.37589×10^{-10}
	0.5	3.49593×10^{-11}	1.43551×10^{-13}	4.02050×10^{-10}
$q=4$	0.1	3.12324×10^{-11}	2.65165×10^{-12}	3.99168×10^{-10}
	0.2	4.34227×10^{-11}	3.67783×10^{-12}	5.43802×10^{-10}
	0.3	4.79300×10^{-11}	4.05153×10^{-12}	5.95169×10^{-10}
	0.4	4.95853×10^{-11}	4.18043×10^{-12}	6.13233×10^{-10}
	0.5	5.01746×10^{-11}	4.21829×10^{-12}	6.19407×10^{-10}

Figure 3: The absolute errors $\alpha=0.1,\ \eta=-0.0025,\ \Delta t=0.0001,\ q=1$

For some values of q and t, discrete L_∞ error norms are recorded for parametes $\alpha=1$, $\eta=1$, $\Delta t=0.0001$ in table 3. Absolute error for $q=1$ at $t=1$ is drawn in Figure 4.

Table 3: L_∞ error norms				
	Time(t)	$l=0$	Various l	[12]
$q=1$	0.2	5.58038×10^{-8}	$1.94765 \times 10^{-10} (p=-0.000319)$	5.55746×10^{-7}
	0.4	8.54479×10^{-8}	1.54361×10^{-9}	9.05507×10^{-7}
	0.6	2.04362×10^{-7}	1.37196×10^{-8}	2.18808×10^{-6}
	0.8	2.80869×10^{-7}	4.72669×10^{-8}	2.93314×10^{-7}
	1.0	2.99588×10^{-7}	1.02753×10^{-7}	3.01455×10^{-6}
$q=2$	0.2	2.82618×10^{-7}	$7.64068 \times 10^{-9} (p=-0.0002930)$	2.56108×10^{-6}
	0.4	4.62302×10^{-7}	2.75394×10^{-8}	4.24308×10^{-6}
	0.6	4.29008×10^{-7}	7.20780×10^{-8}	3.56848×10^{-6}
	0.8	2.56283×10^{-7}	2.07798×10^{-7}	1.46518×10^{-6}
	1.0	8.03168×10^{-8}	3.03531×10^{-7}	5.54230×10^{-6}
$q=4$	0.2	3.98349×10^{-7}	$6.38513 \times 10^{-7} (p=0.000193)$	1.76161×10^{-6}
	0.4	2.64952×10^{-7}	5.11063×10^{-7}	4.17351×10^{-7}
	0.6	1.73948×10^{-8}	1.59753×10^{-7}	2.42401×10^{-6}
	0.8	8.61362×10^{-8}	3.38624×10^{-9}	2.35757×10^{-6}
	1.0	6.63329×10^{-7}	2.41389×10^{-8}	1.44350×10^{-6}

Fig. 3: The absolute errors $\alpha = 1,\ \eta = 1,\ \Delta t = 0.0001,\ q = 1,\ t = 1, l = 0$

Last, the program is rerun with different parameters $\alpha = 0.01,\ 0.001,\ \eta = 1, 10, 100$ time step $\Delta t = 0.0001$, $q = 1$,space step $N = 8$ to make comparison with results of the spectral collocation method the discontinues Galerkin method and cubic B-spline collocation documented in Table 5 at times $t = 0.5$.

Table 4: L_∞ error norm for the solutions of Example 1 at $t=0.5$ for $\alpha=0.001$ and $\alpha=0.0001$						
	η	Present($l=0$)	Various l	[17]	[18]	[19
$\alpha=0.01$	1	2.43664×10^{-11}	$2.95541 \times 10^{-13} (p=0.00034)$	4.6763×10^{-12}	2.8999×10^{-13}	2.4
	10	6.89380×10^{-10}	$1.18782 \times 10^{-12} (p=0.07066)$	6.2529×10^{-12}	3.3184×10^{-13}	1.2
	100	9.32587×10^{-15}	$4.44089 \times 10^{-16} (p=-0.56755)$	8.0269×10^{-12}	2.4225×10^{-13}	1.2
$\alpha=0.001$	1	2.43607×10^{-11}	$4.138911 \times 10^{-13} (p=0.03311)$	4.5374×10^{-12}	2.8821×10^{-13}	2.4
	10	6.87854×10^{-10}		6.0540×10^{-12}	3.3295×10^{-13}	1.2
	100	9.10383×10^{-15}	$8.88178 \times 10^{-16} (p=-0.85953)$	8.1424×10^{-13}	2.4480×10^{-13}	1.2

3.2 Example 2

With initial profile as

$$u(x,0) = \exp(-40x^2)$$

evolution of solution of GBFE is depicted with different parameters α, η and μ for $N = 80$, $\Delta t = 0.001$, $t = 1.5$ in Figure 4-7

Figure 4: $\alpha = 0$, $\eta = 1$ and $\mu = 0.1$

Figure 5: $\alpha = 1$, $\eta = 0.02$ and $\mu = 0.02$

Figure 6: $\alpha = 1$, $\eta = 0.02$ and $\mu = 0.002$

Figure 7: $\alpha = 1$, $\eta = 0.02$ and $\mu = 0.0002$

3.3 Example 3

Consider GBFE with $\eta = 0, q = 1$ and $\alpha = 1$ with initial and boundary conditions

$$u(x, 0) = x(1 - x^2),\ 0 < x < 1$$

and

$$u(0, t) = u(1, t) = 0,\ t \geq 0.$$

In this example, we computed solutions for $\mu = 2^{-2}$ and $\mu = 2^{-6}$ at $t = 0.1$, 0.3, 0.6, 0.9 with step size $\Delta t = 0.001$. The obtained solutions are plotted in Fig. 8-9. Next, we obtained resulst for $\mu = 2^{-2}$, $\mu = 2^{-4}$, $\mu = 2^{-6}$, $\mu = 2^{-8}$ and $t = 0.5$, 0.9, respectively, with step size $\Delta t = 0.001$. We plotted corresponding solution curves in Fig. 10-11.

Figure 8: $\alpha = 1$, $\eta = 0$ and $\mu = 2^{-2}$.

Figure 9: $\alpha = 1$, $\eta = 0$ and $\mu = 2^{-6}$

Computed solutions of Example 3 for different time levels at fixed.μ.

Figure 10: $\alpha = 1$, $\eta = 0$ and $t = 0.5$.

Figure 11: $\alpha = 1$, $\eta = 0$ and $t = 0.9$

Computed solutions of Example 3 for different time levels at fixed.t.

4 Conclusion

The extended B-spline collocation method in space and the Crank-Nicolson method in time were implemented for obtaining explicit solution of the generalized Burgers–Fisher equation. An alternatif algorithm is suggested to solve the GBFE. For first text problem, present method provided lower error than the cubic B-spline quasi-interpolation method. Similar results of the present method are obtained when compared with A spectral domain decomposition approach, local discontinuous Galerkin method and cubic B-spline collocation method when we catch a suitable free parameter which is determined by scanning the predeterming interval with an small space increment, we have achived less error mostly

References

[1]]X L Han and S J Liu, Journal of Computatinal Aided Design and Computer Graphics 15, 576,2003 (in Chinese).

[2] Rubin S. G. , Graves R. A., "Cubic spline approximation for problems in fluid mechanics", Nasa TR R-436,Washington, DC, (1975).

[3] Xu G, Wang GZ (2008) Extended cubic uniform Bspline and -B-spline. Acta Automatica Sin 34, 980–4.

[4] Nur Nadiah Abd Hamid, Ahmad Abd. Majid, and Ahmad Izani Md. Ismail, Extended Cubic B-spline Interpolation Method Applied to Linear Two-Point Boundary Value Problems World Academy of Science, Engineering and Technology International Journal of Mathematical, Computational, Physical, Electrical and Computer Engineering Vol:4, No:2, 2010

[5] Joan Goh,Ahmad Abd. Majid and Ahmad Izani Md. Ismail, A Comparison of Some Splines-Based Methods for the One-dimensional Heat Equation, World Academy of Science, Engineering and Technology 70 2010

[6] N. N. Abd Hamid, A. A. Majid and A. I MD Ismail, An Extended Cubic B-spline Method for Linear Two-Point Boundary Value Problems Sains Malaysiana 40(11)(2011): 1285–1290

[7] J Goh, A A Majid, A I M Ismail, Extended cubic uniform B-spline for a class of singular boundary value problems, Scienceasia,37(2011), 79-82

[8] I. Dag, D. Irk, and M. Sarı, The extended cubic B-spline algorithm for a modified regularized long wave equation, Chin .Phys. B Vol. 22, No.4(2013) 040207 ■

[9] W. K. Zahra , W. A. Ouf , M. S. El-azab, Cubic B-spline collocation algorithm for the numerical solution of Newell Whitehead Segel type equation, Electronic Journal of Mathematical Analysis and Applications Vol. 2(2) July 2014, pp. 81-100.

[10] I Dag, D Irk and M Sari, The extended cubic B-spline algorithm for a modified regularized long wave equation Chin.Phys.B Vol.22,No.4(2013) 040207

[11] D Irk, I. Dağ and M. Tombul, Extended cubic B-spline solution of the advection-diffusion equation, KSCE Journal of Civil Engineering, May 2015, Volume 19, Issue 4, pp 929–934

[12] Zhu, C. G., Kang, W. S., Numerical solution of Burgers-Fisher equation by cubic B-spline quasi-interpolation, Applied Mathematics and Computation, 216, 2679-2686 (2010).

[13] M Tamsir, N. Dhiman, and V. K. Srivastava, Extended modified cubic B-spline algorithm for nonlinear Burgers' equation, beni-suef university journal of basic and applied sciences 5 (2016) 244–254.

[14] R. Mohammadi, Spline solution of the generalized Burgers'Fisher equation,Applicable Analysis 91(12):2189–2215 · December 2012

[15] R.C. Mittal and Amit Tripathi, Numerical solutions of generalized Burgers–Fisher and generalized Burgers–Huxley equations using collocation of cubic B-splines, International Journal of Computer Mathematics, 2014.

[16] Dag I., Ersoy O., Numerical solution of generalized Burgers-Fisher equation by exponential cubic Bspline collocation method, AIP Conference Proceedings 1648 (2015).

[17] A. Golbabai, M. Javidi, A spectral domain decomposition approach for the generalized Burger's–Fisher equation, Chaos, Solitons and Fractals 39 (2009), pp. 385–392.

[18] R. Zhang, X. Yu, and G. Zhao, The local discontinuous Galerkin method for Burger's–Huxley and Burger's–Fisher equations, Appl. Math. Comput. 218 (2012), pp. 8773–8778.

[19] R.C. Mittal and Amit Tripathi, Numerical solutions of generalized Burgers–Fisher and generalized Burgers–Huxley equations using collocation of cubic B-splines, International Journal of Computer Mathematics, 2014.